

LOAN COPY RETURN
AFWL TECHNICAL LIBRARY
KIRTLAND AFB, NM

0062104



TECH LIBRARY KAFB, NM

Advanced Stability Theory Analyses for Laminar Flow Control

Steven A. Orszag

CONTRACT NAS1-15372
FEBRUARY 1980

NASA



0062104

NASA Contractor Report 3233

Advanced Stability Theory Analyses for Laminar Flow Control

Steven A. Orszag
Cambridge Hydrodynamics, Inc.
Cambridge, Massachusetts

Prepared for
Langley Research Center
under Contract NAS1-15372

NASA
National Aeronautics
and Space Administration

**Scientific and Technical
Information Office**

1980

1. Introduction

In this Report, research work done at Cambridge Hydrodynamics, Inc. under Contract No. NAS1-15372 with NASA Langley Research Center is summarized. More detailed expositions of the work described here are given in the publications cited in the References.

In Sec. 2, work done to extend the SALLY computer code (written by CHI with NASA contract support) to treat the propagation of wave packets is explained.

In Sec. 3, work done to extend the SALLY computer code to treat nonlinear, nonparallel compressible flow over LFC wings is described.

2. Wave Packet Analysis for the SALLY Stability Analysis Code

The SALLY computer code was developed for NASA Langley Research Center by CHI to be a 'black-box' stability analyzer for three-dimensional boundary layer flows over LFC wings.¹⁻³ Among the user options offered by SALLY are maximum growth rate determinations at (i) fixed frequency, (ii) fixed wavelength, (iii) fixed frequency, fixed wavelength, (iv) fixed wavelength, fixed propagation angle, (v) fixed frequency, fixed propagation angle. In the present work, SALLY has been extended to allow a wave packet analysis on an LFC wing.

For steady boundary layer flow on an LFC wing, the equations of a wave packet are

$$\frac{d\vec{x}}{dt} = \frac{\partial \omega}{\partial \vec{k}} \quad (1)$$

$$\frac{d\vec{k}}{dt} = - \frac{\partial \omega}{\partial \vec{x}} \quad (2)$$

where \vec{k} is the wave vector of the packet and \vec{x} is the current location on the wing. The required derivatives on the right side of (1)-(2) are computed using the adjoint eigenfunction of the Orr-Sommerfeld equation on the wing. Thus, if the Orr-Sommerfeld equation is written

$$L(\vec{k}, \vec{x}, \omega(\vec{k}, \vec{x}))\psi = 0, \quad (3)$$

where ψ is the eigenfunction, then differentiation with respect to k gives

$$\frac{\partial L}{\partial \vec{k}} \psi + \frac{\partial L}{\partial \omega} \frac{\partial \omega}{\partial \vec{k}} \psi + L \frac{\partial \psi}{\partial \vec{k}} = 0 . \quad (4)$$

Taking the inner product of (4) with the adjoint χ of the eigenfunction ψ gives

$$\frac{\partial \omega}{\partial \vec{k}} = - \frac{(\chi, \frac{\partial L}{\partial \vec{k}} \psi)}{(\chi, \frac{\partial L}{\partial \omega} \psi)} . \quad (5)$$

Similarly, differentiation of (3) with respect to \mathbf{x} gives

$$\frac{\partial L}{\partial \vec{x}} \psi + \frac{\partial L}{\partial \omega} \frac{\partial \omega}{\partial \vec{x}} \psi + L \frac{\partial \psi}{\partial \vec{x}} = 0 \quad (6)$$

so that

$$\frac{\partial \omega}{\partial \vec{x}} = - \frac{(\chi, \frac{\partial L}{\partial \vec{x}} \psi)}{(\chi, \frac{\partial L}{\partial \omega} \psi)} . \quad (7)$$

The modified SALLY code is being used by NASA LaRC personnel to study transition prediction using wave packets instead of maximum amplification rates.

3. Compressible Flow Stability Analysis

The SALLY stability analysis code has been extended to include the effects of compressibility. The equations for linearized disturbances to general three-dimensional compressible boundary layers are given in Refs. 2 and 4. Using the notation of Ref. 4, the equations may be written:

$$\frac{\partial^2}{\partial y^2} \Psi + A \frac{\partial}{\partial y} \Psi + B \Psi = 0 \quad (8)$$

where

$$\Psi = \begin{pmatrix} \alpha u + \beta w \\ v \\ \hat{T} \\ \alpha u - \beta w \end{pmatrix} \quad (9)$$

and A and B are 4×4 complex matrices whose elements are given below. Here α is the x -wavenumber, β is the z -wavenumber, u and w are the amplitudes of x - and z -velocity fluctuations, respectively, \hat{T} is the amplitude of temperature fluctuations, and v is the amplitude of the normal velocity fluctuations. The elements of the A and B matrices are:

$$A_{11} = \frac{1}{\mu} \frac{d\mu}{dT} T' \quad (10)$$

$$A_{12} = -a_2/a_1 \quad (11)$$

$$A_{13} = \frac{1}{\mu} \frac{d\mu}{dT} (\alpha U' + \beta W') \quad (12)$$

$$A_{14} = 0 \quad (13)$$

$$A_{21} = -a_1 a_3 - a_4 \quad (14)$$

$$A_{22} = -a_5 + a_6 - a_7/a_1 \quad (15)$$

$$A_{23} = a_8 - a_9 a_1 \quad (16)$$

$$A_{24} = 0 \quad (17)$$

$$A_{31} = a_{10} \quad (18)$$

$$A_{32} = a_{11}/a_1 \quad (19)$$

$$A_{33} = a_{12} \quad (20)$$

$$A_{34} = a_{13} \quad (21)$$

$$A_{41} = 0 \quad (22)$$

$$A_{42} = 0 \quad (23)$$

$$A_{43} = a_{14} \quad (24)$$

$$A_{44} = a_{15} \quad (25)$$

$$B_{11} = a_{16} + a_4 a_2/a_1 \quad (26)$$

$$B_{12} = a_{17} + a_5 a_2/a_1 \quad (27)$$

$$B_{13} = a_{18} - a_8 a_2 / a_1 \quad (28)$$

$$B_{14} = 0 \quad (29)$$

$$B_{21} = a_7 a_4 / a_1 - a_{19} - a_1 a_{20} - a_4 a_6 \quad (30)$$

$$B_{22} = a_7 a_5 / a_1 - a_{21} - a_1 a_{22} - a_5 a_6 \quad (31)$$

$$B_{23} = - a_7 a_8 / a_1 - a_{23} - a_1 a_{24} + a_8 a_6 \quad (32)$$

$$B_{24} = 0 \quad (33)$$

$$B_{31} = - a_{11} a_4 / a_1 \quad (34)$$

$$B_{32} = a_{25} - a_{11} a_5 / a_1 \quad (35)$$

$$B_{33} = a_{26} + a_{11} a_8 / a_1 \quad (36)$$

$$B_{34} = 0 \quad (37)$$

$$B_{41} = 0 \quad (38)$$

$$B_{42} = a_{27} \quad (39)$$

$$B_{43} = a_{28} \quad (40)$$

$$B_{44} = a_{29} \quad (41)$$

$$a_1 = -i\gamma M_e^2(\alpha U + \beta W - \omega) \quad (42)$$

$$a_2 = \frac{i(\alpha^2 + \beta^2)R}{\mu} - \frac{1}{3}(1+2d)(\alpha^2 + \beta^2)\gamma M_e^2(\alpha U + \beta W - \omega) \quad (43)$$

$$a_3 = -\frac{i}{L} \quad (44)$$

$$a_4 = -i \quad (45)$$

$$a_5 = T'/T \quad (46)$$

$$a_6 = \frac{2i}{3}(2+d)\gamma M_e^2 \frac{1}{L} [(\alpha U + \beta W - \omega) \frac{1}{\mu} \frac{du}{dT} T' + \alpha U' + \beta W' + \frac{T'}{T}(\alpha U + \beta W - \omega)] \quad (47)$$

$$a_7 = -i\gamma M_e^2(\alpha U' + \beta W') \quad (48)$$

$$a_8 = -\frac{i}{T}(\alpha U + \beta W - \omega) \quad (49)$$

$$a_9 = \frac{i}{L} \frac{2}{3}(2+d) \frac{1}{T}(\alpha U + \beta W - \omega) \quad (50)$$

$$a_{10} = 2\sigma(\gamma-1)M_e^2(\alpha U' + \beta W')/(\alpha^2 + \beta^2) \quad (51)$$

$$a_{11} = i \frac{R\sigma}{\mu} (\gamma-1)M_e^2(\alpha U + \beta W - \omega) \quad (52)$$

$$a_{12} = \frac{2}{\kappa} \frac{d\kappa}{dT} T' \quad (53)$$

$$a_{13} = 2\sigma(\gamma-1)M_e^2(\alpha W' - \beta U')/(\alpha^2 + \beta^2) \quad (54)$$

$$a_{14} = \frac{1}{\mu} \frac{d\mu}{dT} (\alpha W' - \beta U') \quad (55)$$

$$a_{15} = \frac{1}{\mu} \frac{d\mu}{dT} T' \quad (56)$$

$$a_{16} = - \frac{iR}{\mu T} (\alpha U' + \beta W' - \omega) - \alpha^2 - \beta^2 \quad (57)$$

$$a_{17} = - \frac{R}{\mu T} (\alpha U' + \beta W') + i(\alpha^2 + \beta^2) \frac{1}{\mu} \frac{d\mu}{dT} T' \quad (58)$$

$$+ \frac{1}{3} i(1+2d) (\alpha^2 + \beta^2) \frac{T'}{T}$$

$$a_{18} = - \frac{1}{3} (1+2d) (\alpha^2 + \beta^2) \frac{1}{T} (\alpha U + \beta W - \omega) \quad (59)$$

$$+ \frac{1}{\mu} \frac{d\mu}{dT} (\alpha U'' + \beta W'') + \frac{1}{\mu} \frac{d^2 \mu}{dT^2} T' (\alpha U' + \beta W')$$

$$a_{19} = 0 \quad (60)$$

$$a_{20} = - \frac{i}{L} \left[\frac{2}{\mu} \frac{d\mu}{dT} T' + \frac{2}{3} (2+d) \frac{T'}{T} \right] \quad (61)$$

$$a_{21} = \frac{T''}{T} - \frac{(T')^2}{T^2} \quad (62)$$

$$a_{22} = \frac{1}{L} \left[-\alpha^2 - \beta^2 + \frac{2}{3} (2+d) \frac{T'^2}{T^2} \frac{1}{\mu} \frac{d\mu}{dT} \right. \quad (63)$$

$$\left. + \frac{2}{3} (2+d) \frac{T''}{T} - \frac{iR}{\mu T} (\alpha U + \beta W - \omega) \right]$$

$$a_{23} = \frac{i}{T} (\alpha U' + \beta W') - \frac{iT'}{T^2} (\alpha U + \beta W - \omega) \quad (64)$$

$$a_{24} = \frac{i}{L} \left\{ \frac{1}{\mu} \frac{d\mu}{dT} (\alpha U' + \beta W') + \frac{2}{3}(2+d) \left[\frac{1}{\mu} \frac{d\mu}{dT} T' (\alpha U + \beta W - \omega) + \frac{1}{T} (\alpha U' + \beta W') \right] \right\} \quad (65)$$

$$a_{25} = - \frac{R\sigma}{\mu} \frac{T'}{T} + i2\sigma(\gamma-1)M_e^2(\alpha U' + \beta W') \quad (66)$$

$$a_{26} = i \frac{R\sigma}{\mu T} (\alpha U + \beta W - \omega) - \alpha^2 - \beta^2 + \frac{T''}{\kappa} \frac{d\kappa}{dT} + \frac{T'^2}{\kappa} \frac{d^2\kappa}{dT^2} + \sigma(\gamma-1)M_e^2 \frac{1}{\mu} \frac{d\mu}{dT} (U'^2 + W'^2) \quad (67)$$

$$a_{27} = - \frac{R}{\mu T} (\alpha W' - \beta U') \quad (68)$$

$$a_{28} = \frac{1}{\mu} \frac{d\mu}{dT} (\alpha W'' - \beta U'') + \frac{1}{\mu} \frac{d^2\mu}{dT^2} T' (\alpha W' - \beta U') \quad (69)$$

$$a_{29} = - \frac{iR}{\mu T} (\alpha U + \beta W - \omega) - \alpha^2 - \beta^2 \quad (70)$$

Here

$$L = \frac{R}{\mu} + i \frac{2}{3}(2+d) \gamma M_e^2 (\alpha U + \beta W - \omega) \quad (71)$$

and U , W are the boundary layer velocity components, T is the temperature, μ is the viscosity coefficient, κ is the thermal conductivity coefficient, σ is the Prandtl number, γ is the ratio of the specific heats, R is the Reynolds number, and d is the ratio of the second to the first viscosity coefficient.

Equations (8)-(71) are to be solved subject to the boundary conditions of vanishing fluctuations at the

boundaries $y = 0, \infty$.

The numerical method used to solve (8)-(71) is explained in the Appendix. A new iteration procedure to solve the Chebyshev-spectral approximation to (8)-(71) has been developed that allows solution with up to 513 Chebyshev modes on the CDC Cyber computers at NASA LaRC. Some variants on the methods described in the Appendix are implemented in the compressible stability code. They are:

- (i) A variable resolution mapping in the y -direction is implemented. Early numerical experiments with the SALLY code indicated that the critical layer of compressible boundary layer flows lies much farther from the wall (wing) than in incompressible flows. In order to resolve adequately the flow region near this critical layer, a variable resolution mapping that gives extra weight to the critical layer is used.
- (ii) In order to minimize the computational expense due to resolution of oscillations of the boundary layer at large distances from the wing, where the boundary layer codes give constant unperturbed flow properties, the exact modes of oscillation of the boundary layer at large distances are determined by an algebraic eigenvalue analysis. The boundary conditions at ∞ are then replaced by conditions at finite y that preclude the presence of growing oscillations at ∞ .

Technical details of this and other matters concerning the compressible SALLY code will be given in a formal paper for publication in the open literature. This paper will be prepared upon completion of numerical calculations of the stability properties of various LFC wings using these compressible stability analysis codes.

4. Nonlinear and Nonparallel Stability Analysis of Compressible Flows

The equations for the nonlinear and nonparallel evolution of three-dimensional disturbances of three-dimensional compressible boundary layers were given in Ref. 2. Even with some notational simplifications, these equations require nearly 40 pages of text to write down. Their direct coding on a computer would be extremely laborious and likely to be subject to serious error. Fortunately, a much easier way to solve the equations required (or, rather, to develop a theory that is nearly the same as that of Ref. 2 that leads to much simpler equations than in Ref. 2) has been developed.

The key idea is to apply the transform methods developed by CHI for solution of the Navier-Stokes equations⁵. In this Section, a simple example of this idea is given postponing full discussion of the technical details of its application to compressible flow until a formal paper on the results is prepared for publication.

Consider the inviscid Burgers' equation

$$\frac{\partial u(x,t)}{\partial t} + u(x,t) \frac{\partial u(x,t)}{\partial x} = 0 \quad (72)$$

with periodic boundary conditions $u(x+2\pi, t) = u(x, t)$. Let us consider perturbations to the steady solution $u(x, t) = U$ of the form

$$\begin{aligned} u = U + \varepsilon \operatorname{Re}(u_1^1(t) e^{ikx}) \\ + \varepsilon^2 \operatorname{Re}(u_2^2(t) e^{2ikx} + u_1^2 e^{ikx}) \\ + \varepsilon^3 \operatorname{Re}(u_1^3 e^{ikx} + \dots) + \dots \quad . \end{aligned} \quad (73)$$

When nonlinear stability theory is applied to this problem, it is necessary to compute the following terms:

$$iUu_1^1$$

$$2iUu_2^2 + i(u_1^1)^2 \quad (74)$$

$$iUu_1^3 + iu_2^2(u_1^1)^*, \text{ etc.}$$

For the present example, the computation of these terms is obviously straightforward and presents no unusual difficulties. However, the equations of compressible stability analysis are considerably more complicated and the calculation of the required terms presents no trivial challenge to the computer.

The new method involves the evaluation of the nonlinear term $u \frac{\partial u}{\partial x}$ using collocation methods at discrete points x_i and ϵ_j using the explicit form (73), neglecting all terms not shown explicitly. Inverse fast Fourier transformation then allows the computation of wavenumbers 1, 2 of the nonlinear term (still as a function of ϵ_j). If the results of this pseudospectral computation are denoted $N(\epsilon_j)$, then the required nonlinear interaction terms in (74) are computed as follows. If terms through order ϵ^3 are required, then N should be evaluated using 4 arbitrary values of ϵ_j . The various coefficients are determined by solving for N_0, N_1, N_2, N_3 from

$$N(\epsilon_j) = N_0 + N_1 \epsilon_j + N_2 \epsilon_j^2 + N_3 \epsilon_j^3 + \dots \quad (75)$$

Choosing $\varepsilon_j = \varepsilon/2^j$ for some small $\varepsilon > 0$ allows the efficient computation of the nonlinear interaction terms N_0, N_1, N_2, N_3 using the formulae for Richardson extrapolation.⁶

Using these ideas, it has been possible to code the nonlinear and nonparallel flow terms for compressible boundary layers very efficiently, with minimal chance for programming error. The computer codes involve just the programming of the nonlinear terms of the compressible Navier-Stokes equations by pseudospectral (collocation) methods and then a final Richardson extrapolation to extract the required information. These new subroutines are being installed in the standard SALLY code during FY80.

Similar ideas to those outlined above should revolutionize numerical methods for nonlinear, nonparallel stability theory in more general problems.

References

1. A.J. Srokowski and S.A. Orszag, "Mass Flow Requirements for LFC Wing Design". AIAA Paper 77-1222 (1977).
2. D.J. Benney and S.A. Orszag, Stability Analysis for Laminar Flow Control. Part I." NASA CR-2910 (1977).
3. S.A. Orszag, "Stability Analysis for Laminar Flow Control. Part II." NASA CR-3249 (1980).
4. L.M. Mack, "On the Stability of the Boundary Layer on a Transonic Swept Wing." AIAA Paper 79-0264 (1979).
5. S.A. Orszag, "Numerical Simulation of Incompressible Flows Within Simple Boundaries. I. Galerkin(Spectral) Representations." Stud. in Appl. Math. 50,293(1971).
6. C.M. Bender and S.A. Orszag, Advanced Mathematical Methods for Scientists and Engineers. McGraw-Hill, New York (1978).

Appendix

1. INTRODUCTION

In this appendix some new techniques are introduced that permit the efficient application of spectral methods to solve problems in (nearly) arbitrary geometries. The resulting methods are a viable alternative to finite difference and finite element methods for these problems. Spectral methods should be particularly attractive for problems in several space dimensions in which high accuracy is required.

Spectral methods are based on representing the solution to a problem as a truncated series of smooth functions of the independent variables. Whereas finite element methods are based on expansions in local basis functions, spectral methods are based on expansions in global functions. Spectral methods are the extension of the standard technique of separation of variables to the solution of arbitrarily complicated problems.

Spectral methods are best introduced for the simple one-dimensional heat equation. Consider the mixed initial-boundary value problem

$$\frac{\partial u(x,t)}{\partial t} = K \frac{\partial^2 u(x,t)}{\partial x^2} \quad (0 < x < \pi, t > 0) \quad (1.1)$$

$$u(0,t) = u(\pi,t) = 0 \quad (t > 0) \quad (1.2)$$

$$u(x,0) = f(x) \quad (0 \leq x \leq \pi) \quad (1.3)$$

The solution to this problem is

$$u(x,t) = \sum_{n=1}^{\infty} a_n(t) \sin nx \quad (1.4)$$

$$a_n(t) = f_n e^{-Kn^2 t} \quad (1.5)$$

where

$$f_n = \frac{2}{\pi} \int_0^{\pi} f(x) \sin nx \, dx \quad (1.6)$$

are the coefficients of the Fourier sign series expansion of $f(x)$.

A spectral approximation to (1.1) - (1.3) is gotten by simply truncating (1.4) to

$$u_N(x, t) = \sum_{n=1}^N a_n(t) \sin nx \quad (1.7)$$

and replacing (1.5) by the evolution equation

$$\frac{da_n}{dt} = -K^2 a_n \quad (n=1, \dots, N) \quad (1.8)$$

with the initial conditions $a_n(0) = f_n$ ($n=1, \dots, N$).

The spectral approximation (1.7) - (1.8) to (1.1) - (1.3) is an exceedingly good approximation for any time t greater than zero as $N \rightarrow \infty$. In fact, the error $u(x, t) - u_N(x, t)$ satisfies

$$u(x, t) - u_N(x, t) = \sum_{n=N+1}^{\infty} f_n e^{-K^2 n^2 t} \sin nx = O(e^{-K^2 t}) \quad (N \rightarrow \infty) \quad (1.9)$$

for any $t > 0$. In contrast to (1.9), finite difference approximations to the heat equation using N grid points in x lead to errors that decay only algebraically with N as $N \rightarrow \infty$. Furthermore, this spectral method for the solution of the heat equation is efficiently implementable by the fast Fourier transform (FFT) in $O(N \log N)$ operations.

There are several significant difficulties in extending the simple spectral method employed for (1.1) - (1.3) to more general problems. These difficulties and their solutions will be discussed in the following sections. Some further details are given in the author's monograph [1]. In Sec. 2, the difficulty caused by nontrivial boundary conditions is discussed. In Sec. 3, the difficulty of treating nonlinear and nonconstant coefficient terms is discussed. Then, in Sec. 4, the properties of spectral methods for problems in simple geometries are summarized. In Sec. 5, the extension of

spectral methods to problems in complicated geometries is explained. In Sec. 6, a new technique for the efficient solution of spectral equations that arise in complicated geometries is given. Some representative test problems are discussed in Sec. 7. Then, in Sec. 8, a summary of results is given together with a glimpse of some other new developments in spectral methods that should find wide application.

2. BOUNDARY CONDITIONS

The Fourier series (1.4) converges fast if $u(x,t)$ is infinitely differentiable and $u(x,t)$ satisfies the boundary conditions

$$\frac{\partial^{2n} u(x,t)}{\partial x^{2n}} = 0 \quad (x = 0, \pi) \quad (2.1)$$

for all nonnegative integers n . Under these conditions, the error after N terms

$$\epsilon_N(x,t) = u(x,t) - \sum_{n=1}^N a_n(t) \sin nx$$

goes to zero uniformly in x faster than any power of $1/N$ as $N \rightarrow \infty$. On the other hand, if $u(x,t)$ is not infinitely differentiable or if any of the conditions (2.1) is violated, then $\epsilon_N(x,t) = O(1/N^p)$ as $N \rightarrow \infty$ for some finite p . For example,

$$1 = \sum_{n=0}^{\infty} (-1)^n \frac{\sin(2n+1)x}{2n+1} \quad (0 < x < \pi), \quad (2.2)$$

but the error incurred by truncating after N terms is of order $1/N$ for any fixed $x, 0 < x < \pi$. Furthermore, the convergence of (2.2) is not uniform in x ; (2.2) exhibits Gibbs' phenomenon, namely

$$\epsilon_N(\xi/N) = O(1) \quad (N \rightarrow \infty, \xi \text{ fixed}).$$

For any fixed N , there are points x at which the error after N terms of (2.2) is not small. The poor convergence of (2.2) is due to the violation of (2.1) for $n = 0$.

More generally, most eigenfunction expansions of a function $f(x)$ converge faster than algebraically (i.e. the error incurred by truncating after N terms goes to zero faster than any finite power of $1/N$ as $N \rightarrow \infty$) only if $f(x)$ is infinitely differentiable and $f(x)$ satisfies an infinite number of special boundary conditions. For example, the Fourier-Bessel expansion

$$f(x) = \sum_{n=0}^{\infty} a_n J_0(\lambda_n x) \quad (0 \leq x \leq 1)$$

where λ_n is the n th smallest root of $J_0(\lambda) = 0$, converges faster than algebraically only if f is infinitely differentiable and

$$\left[\frac{1}{x} \frac{d}{dx} x \frac{d}{dx} \right]^k f(x) = 0 \quad \text{at } x = 1 \quad (2.3)$$

for $k = 0, 1, 2, \dots$.

When a spectral expansion converges only algebraically fast, spectral methods based on these eigenfunction expansions can not offer significant advantages over more conventional (finite difference, finite element) methods. Eigenfunction expansions of this kind should not normally be used unless the boundary conditions of the problem imply all the extra boundary constraints like (2.1) or (2.3). For example, if periodic boundary conditions are compatible with the differential equation to be solved, complex Fourier series are suitable to develop efficient spectral approximations.

In the development of spectral methods for general problems, it is important that the rate of convergence of the eigenfunction expansion being used not depend on special properties of the eigenfunctions, like boundary conditions, but rather depend only on the smoothness of the function being expanded. Of course, if the solution to the problem being solved is not smooth, one should not expect errors that decrease faster than algebraically with $1/N$ when global eigenfunction expansions are used. Faster than algebraic rates of convergence may be achieved for these problems by either patching the solution at discontinuities

(see Sec. 5) or pre-and post-processing of the solution (see [2]).

There is an easy way to ensure that the rate of convergence of a spectral expansion of a function $f(x)$ depends only on the smoothness of $f(x)$, not its boundary properties. The idea is to expand in terms of suitable classes of orthogonal polynomials, including Chebyshev and Legendre polynomials for all those problems in which constraints like (2.1) and (2.3) are unrealistic. These polynomial expansions avoid all difficulties associated with the Gibbs phenomenon provided the solution $f(x)$ is smooth.

From the mathematical point of view, the classical orthogonal polynomials are eigenfunctions of singular Sturm-Liouville problems. It is not hard to show (see [1] for the details) that expansions using eigenfunctions of such singular Sturm-Liouville problems converge at a rate that depends only on the smoothness of $f(x)$, in contrast to eigenfunction expansions based on nonsingular Sturm-Liouville problems that lead to additional boundary constraints like (2.1) on $f(x)$.

These results for orthogonal polynomial expansions are easily demonstrated in the case of Chebyshev polynomial expansions. The n th degree Chebyshev polynomial $T_n(x)$ is defined by

$$T_n(\cos \theta) = \cos n \theta. \quad (2.4)$$

Therefore, if

$$f(x) = \sum_{n=0}^{\infty} a_n T_n(x) \quad (2.5)$$

Then

$$g(\theta) = f(\cos \theta) = \sum_{n=0}^{\infty} a_n \cos n \theta \quad (2.6)$$

Thus, the Chebyshev polynomial expansion coefficients a_n of $f(x)$ are just the Fourier cosine expansion coefficients of the even, periodic function $g(\theta)$. A simple integration by parts argument then shows that

$$n^p a_n \rightarrow 0 \quad (n \rightarrow \infty)$$

provided $g(0)$ (or, equivalently, $f(x)$) has p continuous derivatives. Since

$$|f(x) - \sum_{n=0}^N a_n T_n(x)| \leq \sum_{n=N+1}^{\infty} |a_n| (|x| \leq 1),$$

it follows that the rate of convergence of (2.5) is faster than algebraic if f is smooth.

In summary, spectral expansions should be made using series of orthogonal polynomials unless the boundary conditions of the problem are fully compatible with some other class of eigenfunctions. In practice, Chebyshev and Legendre polynomial expansions are recommended for most applications, supplemented by Fourier series and surface harmonic series when boundary conditions permit.

3. NONLINEAR AND NONCONSTANT COEFFICIENT PROBLEMS

Another difficulty with general kinds of spectral methods is their application to problems with nonlinear and nonconstant coefficient terms. Before explaining the solution to this problem, an illustration of the difficulty is helpful.

Consider the partial differential equation

$$\frac{\partial u}{\partial t} = N(u, u) + Lu \quad (3.1)$$

where $u = u(\vec{x}, t)$ and N is a bilinear (nonlinear) operator that involves only spatial derivatives and L is a linear operator that involves only spatial derivatives. The operators N and L may depend on both \vec{x} and t . A spectral method for the solution of (3.1) is obtained by seeking the solution as a finite spectral expansion:

$$u(\vec{x}, t) = \sum_{n=1}^N a_n(t) \psi_n(\vec{x}) \quad (3.2)$$

where it is assumed for now that $\psi_n(\vec{x})$ ($1 \leq n < \infty$) are a complete set of orthogonal functions. Defining the re-expansion coefficients c_{nmp} and d_{nm} by

$$N(\psi_m, \psi_p) = \sum_{n=1}^{\infty} c_{nmp}(t) \psi_n \quad (3.3)$$

$$L(\psi_m) = \sum_{n=1}^{\infty} d_{nm}(t) \psi_n \quad (3.4)$$

and equating coefficients of $\psi_n(\vec{x})$ ($n=1, \dots, N$) in (3.1), gives

$$\frac{da_n}{dt} = \sum_{m=1}^N \sum_{p=1}^N c_{nmp}(t) a_m(t) a_p(t) + \sum_{m=1}^N d_{nm}(t) a_m(t) \quad (n=1, \dots, N) \quad (3.5)$$

Eqs. (3.5) are the spectral evolution equations for the solution of (3.1). They have one very serious drawback. In general c_{nmp} and d_{nm} are nonzero for typical n, m, p so that evaluation of da_n/dt from (3.5) for all $n=1, \dots, N$ requires $O(N^3)$ arithmetic operations for the bilinear term and $O(N^2)$ operations for the linear term. Thus, solution of (3.5) requires order N^3 operations per time step. Since operational spectral calculations now involve $N \gtrsim 10^6$, the computational cost of the direct solution of (3.5) is prohibitive (even if only linear terms are present).

The problem here is one of computational complexity. Finite difference methods for the solution of (3.1) on N grid points may require only order N operations per time step. If the spectral method really requires order N^3 operations per time step it can not compete when N is large.

Another example illustrating the computational complexity of spectral methods is given by the nonlinear diffusion equation.

$$\frac{\partial u(x, t)}{\partial t} = e^u \frac{\partial^2 u}{\partial x^2}(x, t). \quad (3.6)$$

Expanding the solution as

$$u(x, t) = \sum_{n=1}^N a_n(t) \psi_n(x) \quad (3.7)$$

in terms of the orthonormal functions $\psi_n(x)$, gives

$$\frac{da_n}{dt} = \int \psi_n(x) \exp \left[\sum_{m=1}^N a_m(t) \psi_m(x) \right] \sum_{p=1}^N a_p \psi_p''(x) dx \quad (3.8)$$

for $n = 1, \dots, N$. These evolution equations for $\{a_n(t)\}$ have an exponential degree of computational complexity as they are expressed as an integral functional of $\{a_n(t)\}$.

The solution to the problem of computational complexity is to use the author's transform methods. This technique may be illustrated for a pseudospectral (or collocation) approximation to (3.6) [3]. First, N suitable collocation points x_1, x_2, \dots, x_N lying within the computational domain are introduced. Then, the approximate solution (3.7) is forced to satisfy the partial differential equation (3.6) (or its boundary conditions) exactly at these discrete points at every time t . More specifically, the following three steps are done at each time step t :

(i) Determine N coefficients $a_n(t)$ ($n=1, \dots, N$) so that

$$u(x_j, t) = \sum_{n=1}^N a_n(t) \psi_n(x_j) \quad (j=1, \dots, N). \quad (3.9)$$

(ii) Evaluate $u_{xx}(x_j, t)$ by

$$u_{xx}(x_j, t) = \sum_{n=1}^N a_n(t) \psi_n''(x_j) \quad (j=1, \dots, N). \quad (3.10)$$

(iii) Finally, evaluate $\partial u(x_j, t) / \partial t$ by

$$\frac{\partial u(x_j, t)}{\partial t} = e^{u(x_j, t)} u_{xx}(x_j, t) \quad (j=1, \dots, N) \quad (3.11)$$

and march forward to the next time step.

The idea of the pseudospectral transform method can be restated as follows: Transform freely between physical (x_j) and spectral (a_n) representations, evaluating each term in whatever representation that term is most accurately, and simply, evaluated. Thus, in (3.11), e^u is evaluated in the physical representation while u_{xx} is compared in the spectral representation by (3.9) because it is most accurately done there.

It should be apparent to the reader that pseudospectral transform methods can be applied to any problem that can be treated by finite difference methods regardless of the technical complexity of nonlinear and nonconstant coefficient terms.

What is the computational complexity of pseudospectral transform methods? There are at least three aspects to this question: (i) the computational complexity of differentiation, integration, etc. of spectral series; (ii) the computational complexity of transforming back and forth between physical and spectral representations; and (iii) the computational complexity of solving the resulting equations for the spectral coefficients.

For the expressions of interest, computation of derivatives of an N term spectral expansion requires order N arithmetical operations. For the Fourier series (1.7), this fact is obvious:

$$\frac{d}{dx} \sum_{n=1}^N a_n \sin nx = \sum_{n=1}^N n a_n \cos nx$$

$$\frac{d^2}{dx^2} \sum_{n=1}^N a_n \sin nx = - \sum_{n=1}^N n^2 a_n \sin nx.$$

For the Chebyshev polynomial expansion (2.5), the computational complexity of differentiation is a little less apparent. Since $T_n(\cos \theta) = \cos n \theta$,

$$\frac{T'_{n+1}(x)}{n+1} - \frac{T'_{n-1}(x)}{n-1} = \frac{2}{c_n} T_n(x) \quad (n \geq 0)$$

where $c_0 = 2$, $c_n = 1$ ($n \geq 1$) and $T_0' = T_{-1}' = 0$. Therefore, if

$$\frac{d}{dx} \sum_{n=0}^N a_n T_n(x) = \sum_{n=0}^N b_n T_n'(x),$$

then

$$\begin{aligned} 2 \sum_{n=1}^N a_n T_n'(x) &= \sum_{n=0}^N c_n b_n \left[\frac{T_{n+1}'}{n+1} - \frac{T_{n-1}'}{n-1} \right] \\ &= \sum_{n=1}^{N+1} [c_{n-1} b_{n-1} - b_{n+1}] T_n'(x)/n \end{aligned}$$

Equating coefficients of $T_n'(x)$ for $n = 1, \dots, N+1$ gives the recurrence relation

$$c_{n-1} b_{n-1} - b_{n+1} = 2na_n \quad (1 \leq n \leq N) \quad (3.12)$$

$$b_n = 0 \quad (n > N)$$

The solution of (3.12) for b_n given a_n requires only order N arithmetic operations. Similar recurrence relations can be obtained for differentiation of spectral series based on other sets of orthogonal polynomials and functions.

The computational complexity of transforming between spectral and physical space has several interesting aspects. The problem is: how much computational work is necessary to evaluate

$$u_j = \sum_{n=1}^N a_n \psi_n(x_j) \quad (j=1, \dots, N) \quad (3.13)$$

given $\{a_n\}$ and, inversely, how much work is necessary to compute the expansion coefficients $\{a_n\}$ given $\{u_j\}$? It is obvious that (3.13) can be evaluated for $\{u_j\}$ in $O(N^2)$ operations while it can be solved for a_n in $O(N^3)$

operations. However, these estimates are much too pessimistic; for many important expansions the operation count to perform the transform (3.13) and its inverse is no larger than $O(N(\log N)^p)$ with $p \leq 2$.

In the case of Fourier series, the transform (3.13) and its inverse can be computed in $O(N \log_2 N)$ operations of $N = 2^P$ using the fast Fourier transform. However, most of the computational efficiency of transform methods comes not from the FFT but from the separability of multi-dimensional transforms. Thus, a three-dimensional discrete Fourier transform can be expressed as three one-dimensional Fourier transforms

$$\sum_{j=0}^{J-1} \sum_{k=0}^{K-1} \sum_{\ell=0}^{L-1} a(j, k, \ell) \exp[2\pi i(\frac{jm}{J} + \frac{kn}{K} + \frac{\ell p}{P})] \quad (3.14)$$

$$= \sum_{j=0}^{J-1} e^{2\pi i jm/J} \sum_{k=0}^{K-1} e^{2\pi i kn/K} \sum_{\ell=0}^{L-1} a(j, k, \ell) e^{2\pi i \ell p/L}$$

The left side of (3.14) requires roughly $(JKL)^2$ operations to evaluate at all the points $0 \leq m \leq J, 0 \leq n \leq K, 0 \leq p \leq L$. On the other hand, even without the FFT the right side of (3.14) requires only about $(JKL)(J+K+L)$ operations to evaluate at all the points. When the FFT is applied to the one-dimensional transforms on the right side of (3.14), the number of operations necessary to evaluate (3.14) is reduced further to $(JKL)(\log_2 J + \log_2 K + \log_2 L)$ if J, K, L are powers of 2.

For example, the spectral code CENTICUBE solves the Navier-Stokes equations for three-dimensional incompressible flow with periodic boundary conditions; three-dimensional Fourier series with resolutions up to $128 \times 128 \times 128$ are used to represent the flow. The equations for the Fourier components $\hat{u}(\vec{k}, t)$ of the velocity field involve several convolution sums of the form

$$\sum_{|\vec{p}| < 64} \hat{u}(\vec{p}, t) \hat{u}(\vec{k}-\vec{p}, t).$$

It is not hard to show that direct evaluation of all the required convolution sums on the CRAY-1 computer would require (using optimized code) about 5×10^5 s of computer time at each simulation time step. On the other hand, the CENTICUBE code executes on the CRAY-1 in less than 20s per time step! This speedup by a factor 2.5×10^4 is broken down roughly as follows: a factor 2 for using the FFT instead of an optimized matrix multiply to perform a one-dimensional transform and a factor 10^4 to perform the transforms as a sequence of one-dimensional transforms as in (3.14).

More general transforms can also be implemented efficiently. The author has recently shown that 'fast' transforms of N-term series of Legendre polynomials, surface harmonics, Bessel functions, Jacobi polynomials, Gegenbauer polynomials, etc. can be accomplished in $O(N(\log N)^2/\log \log N)$ arithmetic operations while transforms of series of Hermite and Laguerre polynomials can be accomplished in $O(N(\log N)^2)$ operations. At the present time, these transforms are of only academic interest - as discussed above the speed of transform methods for problems of realistic size owes to the formulation of the problem in terms of separable transforms not the existence of a fast transform.

The third question regarding computational complexity of spectral transform methods concerns the complexity of solving the equations for spectral coefficients. In the case of initial value problems solved by explicit time-stepping methods, the answer is provided by the transform methods discussed above: $O(N(\log N)^p)$ operations are required per time step. The answer to the question is more complicated for the solution of boundary value problems or implicit time-stepping methods for initial value problems.

Spectral approximations to general boundary value problems lead to full $N \times N$ matrix equations for the N expansion coefficients a_n . It would seem that solution of these equations requires $O(N^3)$ arithmetic operations while storage of the matrix requires $O(N^2)$ memory locations. Since typical problems now involve $N \sim 10^6$, the direct solution (or even the direct formulation) of such problems is clearly unworkable now. The solution to this problem is

given in Sec. 6. The solution to the spectral equations requires essentially only $O(N(\log N)^p)$ ($p \leq 2$) operations and only $O(N)$ memory locations. Solution of spectral equations, even though they lead to formally full matrix problems, can be accomplished in little more work and memory than required to solve the simplest finite difference equations!

4. TIME DIFFERENCING AND BOUNDARY LAYERS

Spectral methods based on the classical orthogonal polynomials have another feature that is very attractive for some kinds of problems but leads to difficulties with others. This feature is very high resolution near the boundaries.

For example, the collocation points for Chebyshev polynomial pseudospectral methods for problems on $-1 \leq x \leq 1$ are usually chosen to be $x_j = \cos \pi j/N$ ($j=0,1,\dots,N$). The collocation points x_1 and x_{N-1} are within about $\pi^2/2N^2$ of the boundary points x_0 and x_N , respectively, so that the boundary resolution is $\Delta x = O(1/N^2)$. This leads to extremely good resolution properties of spectral methods for boundary layer problems (see [1] and [4]). While resolution of a problem with a boundary layer of thickness $\epsilon \ll 1$ would require $O(1/\epsilon)$ uniformly spaced grid points, it requires only $O(1/\sqrt{\epsilon})$ terms in the Chebyshev spectral series. [Non-uniform grids should, of course, be used in many of these problems. They can also be implemented efficiently in spectral methods using coordinate transformations.]

The high boundary resolution of spectral methods is not directly useful when problems without boundary layers are to be solved. For example, consider the one-dimensional wave equation

$$u_t + u_x = 0 \quad (-1 \leq x \leq 1, t > 0) \quad (4.1)$$

$$u(-1, t) = f(t) \quad (t > 0) \quad (4.2)$$

$$u(x, 0) = g(x) \quad (-1 \leq x \leq 1). \quad (4.3)$$

Using a Chebyshev polynomial spectral representation

$$u(x, t) = \sum_{n=0}^N a_n(t) T_n(x) \quad (4.4)$$

where $T_n(x) = \cos(n \cos^{-1} x)$, the spectral-tau equations for the solution of (4.1) are [1]:

$$\frac{da_n}{dt} = - \frac{2}{c_n} \sum_{\substack{p=0 \\ p+n \text{ odd}}}^N p a_p \quad (0 \leq n \leq N-1) \quad (4.5)$$

$$\sum_{n=0}^N (-1)^n a_n(t) = f(t) \quad (4.6)$$

where $c_0 = 2$, $c_n = 1$ ($n \geq 1$). The numerical solution of (4.5) - (4.6) in time can be done using any absolutely stable time integration method for ordinary differential equations, such as one of the Adams methods [5]. For an explicit Adams method (Adams-Basforth method), absolute stability in solution of (4.5) - (4.6) requires that

$$\Delta t = 0 \left(\frac{1}{N^2} \right). \quad (4.7)$$

This result may be verified from (4.5) - (4.6) using Gerschgorin's theorem on the distribution of eigenvalues or, more physically, from the classical explicit stability condition $\Delta t = 0(\Delta x)$ with $\Delta x = 0(1/N^2)$. Specifically, the first-order Adams-Basforth method (Euler's method) is stable for the solution of (4.5) - (4.6) provided that

$$\Delta t < \frac{8}{N^2}. \quad (4.8)$$

The time step restrictions (4.7) - (4.8) are qualitatively more severe than those encountered by standard finite difference methods for (4.1) - (4.3). The solution to (4.1) - (4.3) does not exhibit boundary layer structure (unless $g(x)$ or $f(t)$ have non-uniform variation) so a uniform grid may be employed giving the stability criterion $\Delta t = 0(1/N)$ using N grid points. The high boundary resolution of the spectral scheme that leads to the more

stringent requirements (4.7) or (4.8) may seem wasted in this problem. In fact, this high boundary resolution is not completely useless; while high-order accurate stable finite difference schemes for solution of (4.1) - (4.3) on a uniform grid are complicated and require a number of spurious numerical boundary conditions (see, for example, [6]), the infinite-order accurate spectral scheme (4.5) - (4.6) is quite straightforward and requires no spurious boundary conditions to be applied. However, it is also nice to know that the stiffness of the spectral equations can be easily circumvented and time step restrictions like those of finite difference schemes can be easily obtained.

At the present time, there are three alternative ways to avoid the severe time step restrictions of spectral methods in problems that do not exhibit strong boundary-layer structure. First, a semi-implicit scheme [1] may be employed to relax the time step restrictions (4.7) - (4.8) to $\Delta t = 0(1/N)$ as for finite difference schemes. The idea here is to treat implicitly just those parts of the problem, to wit, the boundary regions, that lead to the stiffness of the spectral equations.

Second, it is possible to formulate explicit unconditionally stable time differencing schemes for spectral methods (Turkel & Gottlieb, to be published). Domain of dependence arguments may be used to demonstrate the existence of conditional stability bounds for explicit finite difference methods. These bounds can be avoided in explicit spectral methods because global data is involved at each time step to march the solution forward in time. A simple example of an unconditionally stable explicit spectral method may be given for

$$\frac{\partial u(k,t)}{\partial t} = -iku(k,t) \quad (4.9)$$

which is the equation for a Fourier mode of the solution to $\partial u/\partial t + \partial u/\partial x = 0$ with periodic boundary conditions. Leapfrog time differencing of (4.9) gives

$$\frac{u(k,t+\Delta t) - u(k,t-\Delta t)}{2\Delta t} = -iku(k,t) \quad (4.10)$$

which is second-order accurate in Δt and has the explicit stability bound $|k\Delta t| \leq 1$. However the modified scheme

$$\frac{u(k, t+\Delta t) - u(k, t-\Delta t)}{2\Delta t} = -i \frac{\sin k\Delta t}{\Delta t} u(k, t) \quad (4.11)$$

is still second-order accurate but it is unconditionally stable since $|\sin k\Delta t| \leq 1$ for all $k\Delta t$. [By an accidental cancellation (4.11) happens to give the exact solution $e^{-ik\Delta t}$ of (4.9) for all $k\Delta t$.]

The third method to stabilize time integrations of spectral methods is to use a full implicit time integration method. With N degrees of freedom used to resolve the spatial dimensions, full implicit schemes give full $N \times N$ sets of linear equations to solve at each time step. An efficient method to solve these equations is presented in Section 6.

5. FORMULATION OF SPECTRAL METHODS IN COMPLEX GEOMETRIES

For most problems in complex geometries, it is both inefficient and disadvantageous to seek special sets of spectral expansion functions tuned to the details of the geometry. First, determination of such a set of special functions is itself a computationally difficult problem in the complex geometry that must be repeated for every new geometry. Second, unless very special care is taken. the resulting spectral series will not be guaranteed of fast convergence properties for the problem of interest. Third, much storage will be required for the expansion functions themselves and fast, separable, transforms between physical space and transform space will not normally be available.

There are two very general and powerful methods for the formulation of spectral methods in complex geometries that appear to preserve all the nice properties of spectral methods in simpler geometries, namely, mapping and patching.

The idea of mapping is to transform the complex geometry into a simpler one by a smooth transformation. For example, the annular region

$$f_1(\theta) \leq r \leq f_2(\theta) \quad (5.1)$$

$$0 \leq \theta < 2\pi$$

where (r, θ) are polar coordinates is transformed into the rectangle

$$-1 \leq z \leq 1 \quad (5.2)$$

$$0 \leq \theta < 2\pi$$

by the simple stretching transformation

$$z = 2 \frac{r-f_1(\theta)}{f_2(\theta)-f_1(\theta)} - 1. \quad (5.3)$$

In the mapped coordinates (5.2), a spectral expansion using Chebyshev polynomials in z and Fourier series in θ is appropriate (because the solution to the problem must be periodic in θ). The complication of using a coordinate transformation appears in the coefficients of the differential equation in the transformed domain. For example, derivatives are transformed according to

$$\frac{\partial u}{\partial r} \Big|_{\theta} = \frac{2}{f_2(\theta)-f_1(\theta)} \frac{\partial u}{\partial z} \Big|_{\theta} \quad (5.4)$$

$$\frac{\partial u}{\partial \theta} \Big|_r = \frac{\partial u}{\partial \theta} \Big|_z - \frac{[z(f_2' - f_1') - (f_2' + f_1')]}{f_2' - f_1'} \frac{\partial u}{\partial z} \Big|_{\theta} \quad (5.5)$$

The complication of the equation due to transformation causes no essential difficulty in the implementation of explicit time stepping schemes for initial-value problems because transform methods are still applicable. For boundary value problems and full implicit treatment of initial value problems, it is essential that the full matrix equations be solved by the fast iteration procedures introduced in the next Section.

The simple stretching transformation (5.3) can only be applied if the boundaries $r = f_1$ and $r = f_2$ are single-valued functions of θ . More general boundaries require more sophisticated mappings. In two dimensions, conformal mapping is sometimes useful. Also, pseudo-Lagrangian marker particles may be used to define a coordinate frame. Many other sources of suitable transformations are possible (see, for example, [7]).

The idea of patching is to subdivide a complicated region into a number of simpler regions, make spectral expansions in each of the simpler regions, and then solve the problem in the complicated region by applying suitable continuity conditions across the artificial internal boundaries. For example, consider the Poisson equation

$$\nabla^2 u = f$$

in the L-shaped domain

$$\begin{aligned} 0 \leq x \leq 1, \quad 0 \leq y \leq 1 \\ -1 \leq x \leq 0, 0 \leq y \leq 2. \end{aligned} \tag{5.6}$$

The domain is subdivided into three domains:

$$\text{I: } 0 \leq x \leq 1, 0 \leq y \leq 1 \tag{5.7a}$$

$$\text{II: } -1 \leq x \leq 0, 0 \leq y \leq 1 \tag{5.7b}$$

$$\text{III: } -1 \leq x \leq 0, 1 \leq y \leq 2 \tag{5.7c}$$

In each of these regions, $u(x, y)$ is represented as a double Chebyshev series:

$$u_I(x, y) = \sum_m \sum_n a_{mn}^I T_m(2x-1) T_n(2y-1) \tag{5.8a}$$

$$u_{II}(x, y) = \sum_m \sum_n a_{mn}^{II} T_m(2x+1) T_n(2y-1) \tag{5.8b}$$

$$u_{III}(x,y) = \sum \sum a_{mn}^{III} T_m(2x+1) T_n(2y-3) \quad (5.8c)$$

Across the internal artificial boundaries, continuity of u and the flux of u is imposed. For example, at the interface between regions I and II, it is necessary that

$$u_I(0,y) = u_{II}(0,y) \quad (0 \leq y \leq 1) \quad (5.9a)$$

$$\frac{\partial u_I}{\partial x}(0,y) = \frac{\partial u_{II}}{\partial x}(0,y) \quad (0 \leq y \leq 1). \quad (5.9b)$$

The system of equations in regions I, II, III together with the continuity conditions of the form (5.9) gives a spectrally patched solution to the Poisson equation in the L-shaped domain (5.6).

The advantage of spectral methods over more conventional methods for patched problems is that the spectral schemes require only physical continuity conditions at the internal interfaces. On the other hand, finite difference methods require spurious boundary conditions at interfaces whenever the order of the numerical scheme is higher than that of the differential equation to be solved.

6. SPECTRAL ITERATION METHOD

Consider the solution of a general linear differential equation $Lu = f$. (Extensions to nonlinear problems will be discussed later.) Let an N -term spectral approximation to this problem be given by

$$L_{sp} u_N = f_N \quad (6.1)$$

where f_N is a suitable N -term approximation to f . As mentioned several times earlier the matrix representation of (6.1) is generally a full $N \times N$ matrix so that direct solution of (6.1) by Gauss elimination methods would require order N^2 storage (for the matrix representation of L_{sp}) and order N^3 arithmetic operations.

In this Section, a method will be described that permits solution of (6.1) using order N storage locations with the number of arithmetic operations of order the larger of $N \log N$ and the number of operations required to solve $Lu = f$ by a first-order finite-difference method. The important conclusion is that spectral methods for general problems in general geometries can be implemented efficiently with operation costs and storage not much larger than that of the simplest finite-difference approximation to the problem with the same number of degrees of freedom. Since spectral methods require much less degrees of freedom to achieve a given accuracy or achieve much higher accuracy for a given number of degrees of freedom than required by finite-order finite-difference approximations, important computational efficiencies result from the new method.

The idea of the iteration method is as follows: Suppose that it is possible to construct an approximation L_{ap} to the spectral operator L_{sp} that has the following properties:

- (i) L_{ap} has a sparse matrix representation so that it can be represented using only $O(N)$ storage locations;
- (ii) L_{ap} is efficiently invertible in the sense that the equation

$$L_{ap} u_N = f_N \quad (6.2)$$

is soluble as efficiently as a first-order finite-difference approximation to the problem.

- (iii) L_{ap} approximates L_{sp} in the sense that

$$0 < m \leq \| L_{ap}^{-1} L_{sp} \| \leq M < \infty \quad (6.3)$$

for suitable constants m , M as $N \rightarrow \infty$. Roughly speaking, (6.3) requires that the eigenvalues of $L_{ap}^{-1} L_{sp}$ be bounded from above and below as $N \rightarrow \infty$. Methods to construct suitable operator approximations L_{ap} will be discussed below.

There are several iteration procedures using L_{ap} that converge efficiently to the solution of (6.1). Three of these procedures are described below:

Richardson (Jacobi) iteration

Consider the iteration scheme

$$L_{ap} u_N^{(n+1)} = L_{ap} u_N^{(n)} - \alpha (L_{sp} u_N^{(n)} - f_N). \quad (6.4)$$

If

$$0 < \alpha < \frac{2}{M} \quad (6.5)$$

then

$$u_N^{(n)} \rightarrow u_N \quad (n \rightarrow \infty) \quad (6.6)$$

where u_N satisfies (6.1): $L_{sp} u_N = f_N$. The proof is elementary: If $\epsilon^{(n)} = u_N^{(n)} - u_N$ is the error, then

$$\epsilon^{(n+1)} = \epsilon^{(n)} - \alpha L_{ap}^{-1} L_{sp} \epsilon^{(n)}.$$

Therefore, noting (6.3),

$$\|\epsilon^{(n+1)}\| \leq \max(|1-\alpha m|, |1-\alpha M|) \|\epsilon^{(n)}\| \quad (6.7)$$

so $\|\epsilon^{(n)}\| \rightarrow 0$ as $n \rightarrow \infty$ if (6.5) holds.

The optimal rate of convergence of Richardson's iteration is normally achieved by choosing α to minimize the factor $\max(|1-\alpha m|, |1-\alpha M|)$ appearing in (6.7). This gives

$$\alpha_{opt} = \frac{2}{M+m} \quad (6.8)$$

so

$$\frac{\|\epsilon^{(n+1)}\|}{\|\epsilon^{(n)}\|} \leq \frac{M-m}{M+m}. \quad (6.9)$$

Since, as shown below, it is possible to find L_{ap} such that $M \leq 2.5$ and $m \geq 1$ for nearly arbitrary spectral operators L_{sp} , it follows that Richardson's method decreases the norm of the error in the solution of (6.1) by at least a factor $(5/2+1)/(5/2-1) = 7/3$ at each iteration. (Here $\alpha_{opt} = 4/7$.) The accuracy of a typical initial approximation to u_N is improved by a factor 10^6 after about 16 iterations independent of the resolution N.

How much computational work is required per iteration? The righthand side of (6.4) can be evaluated in $O(N \log N)$ operations by transform methods because $u_N^{(n)}$ is known. Also the solution of (6.4) for $u_N^{(n+1)}$ can be accomplished efficiently because of assumption (ii) above.

Chebyshev iteration

The rate of convergence of the scheme (6.4) can be accelerated using Chebyshev acceleration [8]. The scheme is

$$L_{ap} u_N^{(n+1)} = L_{ap} [\omega_n u_N^{(n)} + (1-\omega_n) u_N^{(n-1)}] - \alpha \omega_n (L_{sp} u_N^{(n)} - f_N) \quad (6.10)$$

where

$$\omega_n = \frac{2\beta T_n(\beta)}{T_{n+1}(\beta)} \quad (6.11)$$

and $\beta = \min(|1-\alpha M|^{-1}, |1-\alpha M|^{-1})$. Here $T_n(\beta)$ is the Chebyshev polynomial of degree n and M, m are given in (6.3). It is not hard to show that the error in the solution of (6.1) decreases by at least the factor $\beta + \sqrt{\beta^2 - 1}$ after each iteration of (6.10).

If $M = 2.5$, $m = 1$, then choosing $\alpha = 4/7$ gives $\beta = 7/3$, so the error decreases by at least $(7+2\sqrt{10})/3 \approx 4.44$ at each iteration. Therefore the error in the solution of (6.1) is decreased by a factor 10^6 after about 9 iterations of (6.10), nearly a factor two faster than the Richardson method (6.4). The penalty of using the Chebyshev acceleration method is that two levels of iterates, $u_N^{(n)}$ and $u_N^{(n-1)}$, must be stored.

Conjugate gradient iterations

For many problems, it is possible to accelerate the convergence of the iteration method for solving (6.1) still further by using the conjugate gradient method [9]. The applicability of this method to the solution of spectral equations will be studied in depth elsewhere. The result is that for general Dirichlet problems for elliptic partial differential equations, the error is decreased by 10^4 after only 3 conjugate gradient iterations and by 10^7 after only seven iterations.

Methods to construct a suitable approximation operator L_{ap} will now be explained. The idea is akin to an idea of D'yakonov [10], but differs in one essential respect. D'yakonov considers the solution of the elliptic partial differential equation

$$Du = \nabla \cdot K(x, y) \nabla u = f \quad (6.12)$$

in some region R with, say, Dirichlet boundary conditions on δR . If

$$0 < K_{\min} \leq K(x, y) \leq K_{\max} < \infty,$$

then D'yakonov proposes the iteration scheme

$$\nabla^2 u^{(n+1)} = \nabla^2 u^{(n)} - \alpha(Du^{(n)} - f) \quad (6.13)$$

where $\alpha = 2/(K_{\min} + K_{\max})$. The error $\|u^{(n)} - u\|$ decreases by at least the factor $(K_{\max} - K_{\min})/(K_{\max} + K_{\min})$ at each iteration of (6.13). D'yakonov's method involves approximating the differential operator by a new, simpler, operator whose spectrum is bounded in terms of the original operator.

L_{ap} is constructed from L_{sp} by changing the discretization operator either in addition to or in place of approximating the differential operator. Thus, L_{ap} is constructed by a suitable low-order finite difference approximation to L .

A simple example is given by the second-order differential equation

$$Lu = f(x)u''(x) + g(x)u'(x) + h(x) = v(x) \quad (0 \leq x \leq 2\pi) \quad (6.14)$$

with periodic boundary conditions $u(x+2\pi) = u(x)$ and $f(x) > 0$. A spectral approximation is approximately sought as the finite Fourier series

$$u(x) = \sum_{|k| < K} a_k e^{ikx}. \quad (6.15)$$

If the Fourier coefficients of $f(x)$, $g(x)$, $h(x)$, $v(x)$ are denoted f_k, g_k, h_k, v_k , respectively, then the spectral (Galerkin) equations for a_k are

$$L_{sp} u = \sum_{\substack{|p| < K \\ |k-p| < K}} [-p^2 f_{k-p} + i p g_{k-p} + h_{k-p}] a_p = v_k. \quad (6.16)$$

Clearly, these equations have, in general, a full matrix representation that requires $O(K^2)$ storage locations and $O(K^3)$ operations to invert.

A suitable approximate operator L_{ap} is constructed using the collocation points $x_j = 2\pi j/N$ ($j=0, 1, \dots, N-1$) where $N = 2K$. In the physical space representation, the finite difference approximation

$$L_{ap} u \Big|_{x_j} = f(x_j) \frac{u_{j+1} - 2u_j + u_{j-1}}{(\Delta x)^2} + g(x_j) \frac{u_{j+1} - u_{j-1}}{2\Delta x} + h(x_j) \quad (6.17)$$

where $u_j = u(x_j)$ and $\Delta x = 2\pi/N$ is used. Obviously, L_{ap} is sparse and efficiently invertible. To verify (6.3) we use the following elementary argument (that may be made more rigorous but no more correct when more involved WKB-like arguments are used.) If λ is an eigenvalue of $L_{ap}^{-1} L_{sp}$ then there exists a function $u(x)$ such that

$$L_{sp} u = \lambda L_{ap} u \quad (6.18)$$

If $u(x)$ is a smooth function of x (in the limit $N \rightarrow \infty$), then both $L_{ap} u$ and $L_{sp} u$ should be good approximations to $Lu(x)$ so (6.18) implies $\lambda \sim 1$. On the other hand, if $u(x)$ is a highly oscillatory function of x (in the limit $N \rightarrow \infty$) then

$$u'' \gg u' \gg u \quad (N \rightarrow \infty). \quad (6.19)$$

Therefore,

$$L_{ap} u \sim f(x_j) \frac{u_{j+1} - 2u_j + u_{j-1}}{(\Delta x)^2} \quad (6.20)$$

and, if transform (pseudospectral) methods are used to evaluate $L_{sp} u$,

$$L_{sp} u \sim f(x_j) \sum_{|k| < K} (-k^2) a_k e^{ikx_j} \quad (6.21)$$

so (6.18) gives

$$f(x_j) \sum_{|k| < K} (-k^2) a_k e^{ikx_j} \sim \lambda f(x_j) \frac{u_{j+1} - 2u_j + u_{j-1}}{(\Delta x)^2} \quad (6.22)$$

The eigenfunctions of (6.22) are

$$u_j = e^{iqx_j} \quad (|q| < K) \quad (6.23)$$

and the associated eigenvalue is

$$\lambda = \frac{(q\Delta x)^2}{4 \sin^2 \frac{1}{2} q \Delta x} \quad (6.24)$$

Since $|q| < K$ with $K = \frac{1}{2} N = \pi/\Delta x$, it follows that

$$1 \leq \lambda \leq \frac{\pi^2}{4} \quad (6.25)$$

Thus, (6.3) holds with $m = 1$ and $M = \pi^2/4 \approx 2.5$.

There are several extensions of the above method for constructing L_{ap} that are important in practice. First, in

the case of Chebyshev-spectral methods, it is appropriate to construct L_{ap}^{-1} using finite-difference approximations based on the collocation points $x_j = \cos \pi j/N$. In this case, the operator bounds (6.3) continue to hold with $M = 2.5$, $m = 1$ for a wide variety of operators L . Second, higher order equations are best treated by writing them as a system of lower order equations. Thus, direct construction of L_{ap}^{-1} for $L = \nabla^4$ gives

$$1 \leq ||L_{ap}^{-1} L_{sp}|| \lesssim 6 \approx \left(\frac{\pi}{4}\right)^2 \quad (6.26)$$

However, introducing $v = \nabla^2 u$, defining the second-order operator K by

$$K_{v}^{(u)} = \frac{\nabla^2 u - v}{\nabla^2 v}, \quad (6.27)$$

and direct construction of K_{ap}^{-1} as a finite-difference operator gives

$$1 \leq ||K_{ap}^{-1} K_{sp}|| \lesssim 2.5. \quad (6.28)$$

Third, odd-order operators, initial-value problems, and problems of mixed type are best treated by constructing L_{ap}^{-1} on a grid that is roughly 50% finer than that used in construction of L_{sp} by collocation. In this case the spectral bounds (6.3) with $M \leq 2.5$ continue to hold for most problems. For example, the operator $\frac{\partial}{\partial x}$ with periodic boundary conditions has spectrum ik while its centered finite-difference approximation has spectrum $i \sin(k\Delta x)/\Delta x$ so

$$||L_{ap}^{-1} L_{sp}|| = O(k\Delta x / \sin k\Delta x)$$

which is unbounded for $|k\Delta x| < \pi$, but bounded by $4\pi/3\sqrt{3} \approx 2.4$ if $|k\Delta x| < 2\pi/3$.

7. EXAMPLES

Consider the solution of the heat equation

$$\frac{\partial u}{\partial t} = \nabla^2 u - e^x - e^y \quad (x, y) \in D \quad (7.1)$$

with Dirichlet boundary conditions

$$u(x,y) = e^x + e^y \quad (x,y) \in \partial D, \quad (7.2)$$

where D is the annular region plotted in Fig. 1 whose inner and outer boundaries are, respectively, (in polar coordinates)

$$r = f_1(\theta) = 0.3 + 0.1 \sin \theta + 0.15 \sin 5\theta \quad (7.3)$$

and

$$r = f_2(\theta) = 1 + 0.2 \cos \theta + 0.15 \sin 4\theta. \quad (7.4)$$

This problem is solved spectrally using the stretching transformation (5.3) to transform D into the rectangular domain (5.2) and then using Fourier series in θ and Chebyshev series in z to represent $u(x,y)$.

As $t \rightarrow \infty$, the solution to (7.1) - (7.4) approaches the steady state solution

$$u(x,y) = e^x + e^y \quad (x,y) \in D \quad (7.5)$$

In Table 1, the maximum pointwise errors in the spectral solution of this problem as $t \rightarrow \infty$ are given. Evidently the rapid convergence of the spectral solution to the exact steady state is preserved in this complex geometry problem. It is not very surprising that, for a given total number of retained modes, the best accuracy is achieved with about 4 times more 'angular' (θ) than 'radial' (z) modes; after all, the 'annulus' (7.3) - (7.4) is on average about π times longer in circumference than in radius.

Another example of the technique suggested in Sec. 6 is the solution of the boundary-layer equations

$$\frac{\partial^3 f}{\partial \eta^3} + f \frac{\partial^2 f}{\partial \eta^2} + \beta(\xi) [1 - (\frac{\partial f}{\partial \eta})^2] = 2\xi [\frac{\partial f}{\partial \eta} \frac{\partial^2 f}{\partial \xi \partial \eta} - \frac{\partial^2 f}{\partial \eta^2} \frac{\partial f}{\partial \xi}] \quad (7.6)$$

where $f(\xi, \eta)$ is the dimensionless stream function and is subject to the boundary conditions

$$f(\xi, 0) = \frac{\partial f}{\partial \eta}(\xi, 0) = 0 \quad (7.7)$$

$$\frac{\partial f}{\partial \eta}(\xi, \eta) \rightarrow 1 \quad (\eta \rightarrow \infty) \quad (7.8)$$

Here (ξ, η) are the Levy-Lees coordinates; ξ increases in the free-stream direction and η increases away from the wall. The non-self-similar solution of these equations for Howarth's flow in which the pressure-gradient parameter $\beta(\xi)$ is given by

$$\beta(\xi) = \frac{\xi}{\xi-4} \quad (7.9)$$

is a standard test problem [11]. Eqs. (7.6) - (7.9) are solved using Chebyshev series in the transformed variable

$$s = \frac{2\eta}{R} - 1 \quad (-1 \leq s \leq 1),$$

where R is chosen so $\eta = R$ is in the free stream and a Crank-Nicolson scheme is used in ξ . The finite-difference errors in ξ are reduced by Richardson extrapolation [13]. The results for the viscous wall stress at two downstream locations near the separation point $\xi \approx 0.901$ are given in Table 2 together with some corresponding finite difference results from Ref. 13. These results illustrate the high accuracy of the spectral schemes with modest resolution.

8. OTHER DEVELOPMENTS

In this paper, techniques have been explained to extend spectral methods efficiently to problems in complex geometries. Many applications of these techniques are underway and results will be presented elsewhere.

There have been several other recent developments of spectral methods that relate to these applications but have independent interest. First, Lagrangian spectral methods have been developed for the solution of high speed flows. In this case, Lagrangian marker particles are used to provide the coordinate transformations necessary for the methods introduced in this paper. Second, fast conformal mapping techniques have

been developed. These techniques are particularly attractive for the solution of free-surface flow problems by spectral methods.

References

1. D. Gottlieb and S. A. Orszag, Numerical Analysis of Spectral Methods, NASA CR-15778 (1977).
2. A. Majda, J. McDonough and S. Osher, "The Fourier method for non-smooth initial data", Math. Comp. 32, 1041 (1978).
3. S. A. Orszag, "Numerical simulation of incompressible flows within simple boundaries I: Galerkin (spectral) representations," Studies in Appl. Math. 50, 293 (1971).
4. S. A. Orszag and M. Israeli, "Numerical simulation of viscous incompressible flows," Ann. Rev. Fluid Mech. 6, 281 (1974).
5. C. W. Gear, Numerical Initial Value Problems in Ordinary Differential Equations, Prentice-Hall, Englewood Cliffs N. J. (1971).
6. J. Oliger, "Fourth order difference methods for the initial boundary value problem for hyperbolic equations," Math. Comp. 28, 15 (1974).
7. J. F. Thompson, F. C. Thames and C. W. Mastin, "Boundary-fitted curvilinear coordinate systems for solution of partial differential equations on fields containing any number of arbitrary two-dimensional bodies," NASA CR-2729 (1977).
8. R. S. Varga, Matrix Iterative Analysis, Prentice-Hall Englewood Cliffs, N.J. (1962).
9. M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear system," J. Res. Nat. Bur. Standards 49, 409 (1952).
10. E. G. D'yakonov, "On an iterative method for the solution of finite difference equations," Dokl. Akad. Nauk SSSR 138, 522 (1961).
11. H. B. Keller and T. Cebeci, "Accurate numerical methods for boundary layer flows. I. Two dimensional laminar flows," in Proc. Second Intl. Conf. on Num. Meth. in Fluid Dynamics, Springer-Verlag, Berlin (1971), p.92.

Table 1. Errors in Steady State Solution of (7.1)-(7.4)

<u>Number of angular modes (θ)</u>	<u>Number of 'radial' Chebyshev modes (z)</u>	<u>Maximum relative error</u>
16	4	1.4×10^{-2}
32	8	2.8×10^{-5}
64	16	2.5×10^{-10}

Table 2. Viscous Wall Stresses for Howarth's Flow

Method	Truncation R \leq s \leq R	Number of grid points (modes) in η	Number of grid points in ξ	Total number of points	A	B
Spectral	8	23	6	138	0.21239	0.04454
Spectral	8	23	11	253	0.20913	0.03471
Spectral with Richardson extrapolation in ξ	8	23	{6,11}	253	0.20718	0.03064
Finite difference [13]	6	121	51	6171	0.20723	0.03083
Finite difference with Richardson extrapolation in ξ and/or η [13]	6	19	16	489	0.20710	0.03171
	6	61	16	1708	0.20741	0.03121
	6	121	51	9282	0.20714	0.03053

$$A = \frac{\partial^2 f}{\partial \eta^2} (\xi = 0.7, \eta = 0) \approx 0.20724$$

$$B = \frac{\partial^2 f}{\partial \eta^2} (\xi = 0.894, \eta = 0) \approx 0.03072$$

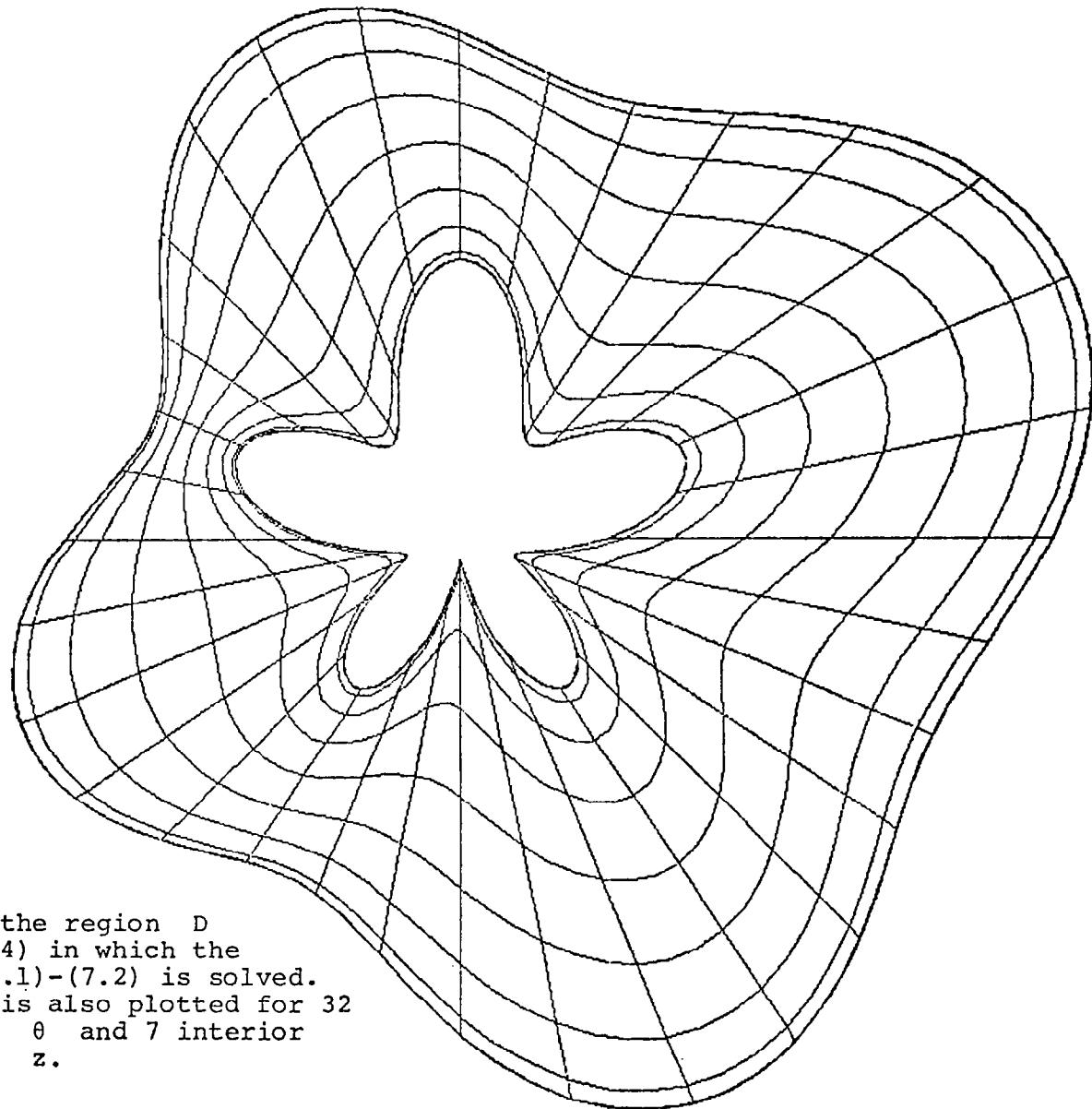


Figure 1. A plot of the region D defined by (7.3)-(7.4) in which the diffusion equation (7.1)-(7.2) is solved. The collocation grid is also plotted for 32 collocation points in θ and 7 interior collocation points in z .

1. Report No. NASA CR-3233	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle ADVANCED STABILITY THEORY ANALYSES FOR LAMINAR FLOW CONTROL		5. Report Date February 1980	
7. Author(s) Steven A. Orszag		6. Performing Organization Code	
9. Performing Organization Name and Address Cambridge Hydrodynamics, Inc. P. O. Box 249 M.I.T. Station Cambridge, MA 02139		8. Performing Organization Report No.	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546		10. Work Unit No.	
15. Supplementary Notes Langley technical monitor was Jerry N. Hefner. Final Report		11. Contract or Grant No. NAS1-15372	
16. Abstract Recent developments of the SALLY computer code for stability analysis of laminar flow control wings are summarized. Extensions of SALLY to study three-dimensional compressible flows, nonparallel and nonlinear effects are discussed.			
17. Key Words (Suggested by Author(s)) Stability theory Laminar flow control Numerical methods Wave packet analysis		18. Distribution Statement Unclassified - Unlimited Subject Category 34	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 48	22. Price* \$4.50

* For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1980